AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior versions and listings of claims in the application:

1. (Canceled)

2. (Currently Amended) An arithmetic apparatus incorporated in a LSI for performing a long integer product-sum arithmetic operation, the arithmetic apparatus comprising:

an arithmetic unit comprising:

an integer based ~~unit arithmetic~~ multiplier circuit;

a finite field $GF(2^m)$-based ~~$GF(2^m)$ based unit arithmetic~~ multiplier circuit logically adjacent to but separated from said integer based ~~unit arithmetic~~ multiplier circuit; [[and]]

an adder circuit shared by the separated integer based multiplier circuit and the finite field $GF(2^m)$-based circuit and configured to operate on data from either the integer based multiplier circuit or the finite field $GF(2^m)$-based circuit; and

a selector configured to select one of said integer ~~unit arithmetic~~ multiplier circuit and said finite field $GF(2^m)$-based ~~$GF(2^m)$ based unit arithmetic~~ multiplier circuit, and

a controller controlling said selector to make said selection.

3. (Previously Presented) An apparatus according to claim 2, wherein the arithmetic unit further comprises:

an adder circuit which has a buffer for storing interim result data, adds the interim result data to result data from one of said integer unit arithmetic circuit and said finite field GF(2^m) based unit arithmetic circuit which is selected by said selector, propagates a carry in an integer based unit arithmetic operation, and propagates no carry in a finite field GF(2^m) based unit arithmetic operation.

4. (Previously Presented) An apparatus according to claim 3, wherein the arithmetic unit further comprises:

a carry holder for storing a carry obtained in a previous operation cycle, and an output-stage adder circuit configured to add the carry in said carry holder to an output from said adder circuit, output an upper bit of an addition result as an updated carry to said carry holder, and output a lower bit of the addition result as operation result data.

5. (Previously Presented) A crypto processing apparatus for selectively encrypting or decrypting based on an integer based operation by said arithmetic apparatus defined in claim 2, and encrypting or decrypting based on a finite field GF(2^m) based unit arithmetic operation by said arithmetic apparatus.

6. (Currently Amended) An arithmetic apparatus incorporated in a LSI, comprising:

an arithmetic unit including an integer unit arithmetic circuit, wherein the arithmetic circuit comprises a full adder, the full adder including a carry propagation section configured to propagate a carry of an operation result of the full adder upon reception of a selection signal corresponding to an integer-based unit arithmetic operation and to not propagate the carry of the full adder upon reception of a selection signal corresponding to a finite field $GF(2^m)$-based unit arithmetic operation; and

a controller configured to output, to said integer unit arithmetic circuit, a selection signal for selecting one of an integer unit arithmetic operation and finite field $GF(2^m)$-based ~~$GF(2^m)$ based~~ unit arithmetic operation; ~~and~~

~~a carry propagation controller configured to propagate, when a long product-sum operation is to be executed, a carry of an operation result obtained by said integer-based unit arithmetic circuit upon reception of a selection signal corresponding to an integer based unit arithmetic operation, and propagate no carry of the operation result upon reception of a selection signal corresponding to a finite field $GF(2^m)$ based unit arithmetic operation,~~

wherein an integer-based ~~integer based~~ multiply operation [[and]] or a finite field $GF(2^m)$-based ~~$GF(2^m)$ based~~ multiply operation is ~~switched~~ selected by controlling the carry propagation.

7. (Original) An apparatus according to claim 6, wherein said integer unit arithmetic circuit comprises a full adder (FA), and said carry propagation controller comprises a switch to which the selection signal and a carry out signal are input, and performs carry propagation control of said full adder in units of bits.

8. (Original) An apparatus according to claim 6, wherein said integer unit arithmetic circuit comprises a fuller adder, and said carry propagation controller comprises a selection section configured to switch between outputting a 2-input EX-OR result obtained by said full adder in units of bits and outputting an EX-OR result based on the result and an input carry as an addition result.

9. (Previously Presented) An apparatus according to claim 6, wherein said integer based unit arithmetic circuit adds by propagating a carry when executing the integer based multiply operation, and adds without propagating any carry when executing the finite field $GF(2^m)$ based multiply operation.

10. (Previously Presented) A crypto processing apparatus for selectively encrypting or decrypting based on an integer based operation by said arithmetic apparatus defined in claim 6, and encrypting or decrypting based on a finite field $GF(2^m)$ based arithmetic operation by said arithmetic apparatus.

11. (Currently Amended) An arithmetic apparatus for long product-sum operation incorporated in a LSI, comprising:

an arithmetic unit module including an integer unit arithmetic circuit ~~a long product-sum operation circuit which executes only polynomial multiplication with a finite field $GF(2^m)$ based polynomial base expression~~ configured to perform one of an integer unit arithmetic operation and a finite field $GF(2^m)$-based unit arithmetic operation; and

a controller module configured to divide the processing of modular multiplication of the integer unit arithmetic operation and the finite field GF(2^m)-based unit arithmetic operation into polynomial multiply processing and [[a]] modulo processing and to cause ~~said long product-sum operation circuit~~ the integer unit arithmetic circuit to execute the polynomial ~~multiplication.~~ multiply processing;

wherein the controller module selects the one of an integer unit arithmetic operation and finite field GF(2^m)-based unit arithmetic operation and executes the modulo processing on the integer unit arithmetic circuit.

12. (Previously Presented) An apparatus according to claim 11, wherein said long product-sum operation circuit comprises a single precision multiplier circuit configured to multiply polynomial data of the finite field GF(2^m) based polynomial base without propagating any carry, and a double precision adder circuit configured to add using a multiply result obtained by said multiplier circuit, and said controller controls said multiplier circuit and said adder circuit in the multiply processing.

13. (Previously Presented) An apparatus according to claim 12, wherein said controller module comprises a quotient acquisition circuit configured to set, in the modulo, a multiply result of two polynomial data as first dividend polynomial data, set predetermined modulo polynomial data as divisor polynomial data, calculate a quotient on the basis of the first or subsequent dividend polynomial data and the divisor polynomial data, and acquire 1-block quotient polynomial data with the number of bits corresponding to a bus width from an upper order, and

when 1-block quotient polynomial data is acquired in the modulo, said multiplier circuit and said adder circuit calculate next dividend polynomial data by subtracting a multiply result of the 1-block quotient polynomial data and the divisor polynomial data from current dividend polynomial data, and said controller module controls said quotient acquisition circuit to repeat processing up to calculation of the dividend polynomial data, to obtain residue data.

14. (Original) An apparatus according to claim 13, wherein in the quotient calculation, said quotient acquisition circuit multiplies inverse data of upper two blocks of the divisor polynomial data and upper two blocks of current dividend polynomial data, and sets a second upper block of the multiply result as the 1-block quotient polynomial data.

15. (Original) An apparatus according to claim 14, wherein said quotient acquisition circuit calculates inverse data from upper two blocks of the divisor polynomial data and stores the data in a memory when acquiring quotient polynomial data in a first operation, and reads out inverse data from said memory when acquiring quotient polynomial data in second and subsequent operations.

16. (Original) An apparatus according to claim 14, wherein in calculating the inverse data, said quotient acquisition circuit counts the number of consecutive 0s from an upper order of upper two blocks of the divisor polynomial data, extracts polynomial data of one block + 1 bit from an upper order such that a most significant bit is set to 1,

obtains an inverse of the extracted polynomial data, obtains 2-block data as a whole by concatenating 1-block corrected data whose least significant bit is 1 and other bits are 0 to a most significant bit of the obtained inverse, and setting, as the inverse data, a result obtained by bit-shifting the data toward an upper order side by the count of 0s.

17. (Previously Presented) A crypto processing apparatus for encrypting or decrypting based on a finite field $GF(2^m)$ based modular multiplication by said arithmetic apparatus defined in claim 11.

18. (Previously Presented) An apparatus according to claim 11, wherein said product-sum operation circuit includes a unit arithmetic circuit, operates said unit arithmetic circuit by propagating a carry when executing an integer based unit arithmetic operation, and operates said unit arithmetic circuit without propagating any carry when executing a finite field $GF(2^m)$ based unit arithmetic operation.

19. (Previously Presented) An arithmetic apparatus comprising:

an arithmetic unit including a long product-sum operation circuit which executes a modular multiplication with a finite field $GF(2^m)$ based polynomial base expression; and

a controller configured to divide the modular multiplication into multiply processing and a modulo and causing said long product-sum operation circuit to execute the modular multiplication,

wherein said long product-sum operation circuit comprises a single precision multiplier circuit configured to multiply polynomial data of the finite field $GF(2^m)$ based

polynomial base without propagating any carry, and a double precision adder circuit configured to add using a multiply result obtained by said multiplier circuit, and said controller controls said multiplier circuit and said adder circuit in the multiply processing,

said controller comprising a quotient acquisition circuit configured to set, in the modulo, a multiply result of two polynomial data as first dividend polynomial data, set predetermined modulo polynomial data as divisor polynomial data, calculate a quotient on the basis of the first or subsequent dividend polynomial data and the divisor polynomial data, and acquire 1-block quotient polynomial data with the number of bits corresponding to a bus width from an upper order; and when 1-block quotient polynomial data is acquired in the modulo, said multiplier circuit and said adder circuit calculate next dividend polynomial data by subtracting a multiply result of the 1-block quotient polynomial data and the divisor polynomial data from current dividend polynomial data, and said controller controls said quotient acquisition circuit to repeat processing up to calculation of the dividend polynomial data, thereby obtaining residue data,

wherein in the quotient calculation, said quotient acquisition circuit multiplies inverse data of upper two blocks of the divisor polynomial data and upper two blocks of current dividend polynomial data, and sets a second upper block of the multiply result as the 1-block quotient polynomial data, and

wherein in calculating the inverse data, said quotient acquisition circuit counts the number of consecutive 0s from an upper order of upper two blocks of the divisor polynomial data, extracts polynomial data of one block + 1 bit from an upper order such that a most significant bit is set to 1, obtains an inverse of the extracted polynomial data,

obtains 2-block data as a whole by concatenating 1-block corrected data whose least significant bit is 1 and other bits are 0 to a most significant bit of the obtained inverse, and setting, as the inverse data, a result obtained by bit-shifting the data toward an upper order side by the count of 0s.